

# ReaderLiSh

„Cartea reflectă ca o oglindă lungul șir de secole al vieții omenirii, istoria luptei sale pentru existență, pentru un viitor mai luminos, suferințele, bucuriile, înfrângerile și biruințele sale toate. Iubiți cartea, îngrijiți-o și citiți cât mai mult. Cartea ne este prieten credincios, de nădejde.”

(G. F. Morozov)

## Motivație

Sunt o amatoare a cărților, în special a celor de dezvoltare personală. Consider că niciodată nu ar trebui să ne oprim din citit, acesta fiind principalul stimulent al creierului. Știind că nu sunt singura care împărtășește această idee, m-am gândit la o aplicație care să vină în ajutorul celor care doresc să citească și în momentele în care nu își pot lua o carte cu ei. După cum știm, tehnologia a avansat destul de mult, iar telefonul a devenit indispensabil în zilele noastre. Așadar, aplicația va fi destinată celor care dispun de un dispozitiv mobil cu Android.

ReaderLiSh este un concept de aplicație pentru telefoanele cu Android care are mai multe funcționalități. În primul rând aplicația permite citirea cărților pe care tu singur le încarci în baza de date, sau a celor deja existente, adăugate de administrator. În al doilea rând, aplicația permite cu ajutorul GPS-ului găsirea de biblioteci și librării în zona în care te afli, sau o altă selectată de utilizator. Ceea ce aduce nou aplicația este una dintre opțiunile în care îți poți cauta o carte după o anumită stare de spirit. Dacă utilizatorul dorește într-un moment de supărare să citească o carte care să îi ridice moralul și să îi îmbunătățească starea de spirit, va putea foarte ușor să scrie sentimentul “fericire” sau “bucurie” sau orice alt astfel de sentiment, iar aplicația îi va returna o serie de cărți care fac parte din categoria aleasă. Pe lângă aceste funcționalități, ReaderLiSh va funcționa și ca un fel de rețea de socializare. Citind o carte, un utilizator îi poate acorda un calificativ (Like) și o poate recomanda și celorlalți utilizatori care folosesc această aplicație.

Toate aceste facilități le-am îmbinat într-un singur concept, ReaderLiSh, care se traduce prin Read, Like & Share.

## Context

Un prim argument în ceea ce privește rolul lecturii în dezvoltarea personală este efectul pe care-l exercită o carte asupra unui individ: prin lecturarea unei cărți, omul își îmbogățește lumea interioară prin intermediul imaginației. Desigur, există și multe filme bazate pe anumite cărți, însă avantajul citirii unei opere este că fiecare își poate imagina într-un anumit fel un personaj, un loc sau poate să judece în felul său un eveniment.

Un al doilea argument în favoarea lecturii este faptul că aceasta oferă subiecte de meditație și reprezintă o sursă imensă de idei, idei care pot fi împărtășite altor oameni sau chiar aplicate în viața de zi cu zi. Astfel, un anumit paragraf dintr-o carte de psihologie poate să reprezinte un răspuns la o întrebare care demult sălășuiește în mintea individului. Ideea principală a unui roman poate fi privită din mai multe puncte de vedere, la fel și acțiunile protagonistului sau chiar decorul, iar aceste aspecte ale unei cărți pot fi discutate la infinit cu un om ce a citit la rândul său opera dată.

În concluzie, consider că lectura este un aspect important în viața individului, indiferent de forma pe care o are: carte, revistă, ziar, carte audio. În dezvoltarea personală aceasta joacă rolul de sursă de informație, ajută la antrenarea imaginației și reprezintă o formă de socializare și descoperire a oamenilor ce împart aceeași pasiune.<sup>i</sup>

## Obiectiv

Îmi propun realizarea unei aplicații care să vină în ajutorul persoanelor pasionate de citit, persoane care doresc să-și îmbunătățească cultura generală prin lecturarea unor cărți de pe un dispozitiv mobil. Aplicația este creată pentru cei care nu dispun de foarte mult timp liber și vor să îl investească într-un mod cât mai inteligent.

## Cerințe funcționale

În momentul în care un utilizator își va descarca și instala aplicația, își va crea un cont în baza de date care va fi personalizat în funcție de fiecare.

Principalul scop al aplicației este citirea cărților, asadar, utilizatorii vor putea citi cărțile din baza de date, sau le vor putea citi pe cele încărcate de ei.

Utilizatorii vor putea face și mici adnotări asupra textului, și anume ei vor putea sublinia un paragraf, vor putea pune un semn la o anumită pagina și desigur să selecteze și anumite fragmente din carte care li se par mai interesante, pentru ca mai apoi să le poată găsi cu ușurință.

Odata activat GPS-ul telefonului, aplicația va avea activă opțiunea de a căuta anumite cărți în bibliotecile și librăriile din zona în care se afla, sau o altă zonă setată de utilizator.

Fiecare carte citită va putea primi și un calificativ (Like), sau nu va primi nimic, depinde de utilizator.

Utilizatorii care citesc anumite cărți și sunt mulțumiți, sau nu, de informațiile pe care le-au găsit, vor putea împărtăși impresiile și cu ceilalți utilizatori ai aplicației. De asemenea ei pot face și anumite recomandări în privința cărților

.

## Platformă de dezvoltare:

- Platforma de dezvoltare Android

În prezent, Android este una din cele mai renumite platforme de dezvoltare oferind posibilitatea realizării de aplicații pentru milioane de dispozitive din întreaga lume.

Rolul dezvoltatorilor de aplicații mobile s-a schimbat dramatic în ultimii ani și au ajuns să fie foarte căutați pe piața de muncă. Producătorii de telefoane, furnizorii de platforme și chiar și operatorii de rețea se află în competiție pentru realizarea celei mai mari comunități de dezvoltare.

- SDK-ul Android

SDK-ul Android conține un set de instrumente de dezvoltare, precum biblioteci, program de depanare, un emulator de dispozitiv, documentație, mostre de cod și tutoriale menite să simplifice dezvoltarea de aplicații mobile. Platformele de dezvoltare sprijinite în prezent înglobează calculatoare pe X86 care rulează Linux, Mac OS sau mai recent Windows 7 sau 8.1.4

Îmbunătățirile aduse la SDK-ul Android facilitează dezvoltarea de aplicații destinate primelor versiunii ale platformei Android, astfel se pot realiza aplicații și pentru dispozitivele mai vechi. Instrumentele pentru dezvoltare pot fi descărcate în funcție de platforma și versiunea dorită pentru a se asigura compatibilitatea acestora cu dispozitivele pentru care sunt realizate.

## Tehnologii folosite:

Pentru dezvoltarea aplicației “ReaderLiSh ” au fost folosite următoarele tehnologii:

- Sistemul de operare Android
- Google Maps API
- Server Java
- Fire de execuție
- SQLite

## Sistemul de operare Android

*Android* este un sistem de operare pentru dispozitive și telefoane mobile bazată pe nucleul Linux, dezvoltată inițial de compania Google, iar mai târziu de consorțiul comercial Open Handset Alliance. Android permite dezvoltatorilor să scrie cod gestionat în limbajul Java, controlând dispozitivul prin intermediul bibliotecilor Java dezvoltate de Google. Aplicații scrise în C și în alte limbaje pot fi compilate în cod mașină ARM și executate, dar acest model de dezvoltare nu este sprijinit oficial de către Google. Lansarea platformei Android, la 5 noiembrie 2007, a fost anunțată prin fondarea Open Handset Alliance, un consorțiu de companii de

hardware, software și de telecomunicații, consacrat dezvoltării de standarde deschise pentru dispozitive mobile. Google a lansat cea mai mare parte a codului Android sub licența Apache, o licență de tip free-software și open-source.

Începând cu 21 octombrie 2008, Android a fost disponibil ca *Open Source*. Google a deschis întregul cod sursă, care anterior era disponibil sub licența Apache. Aceasta permite producătorilor să adauge extensii proprietare, fără a le face disponibile comunității open source. În timp ce contribuțiile Google la această platformă se așteaptă să rămână open source, numărul versiunilor derivate ar putea exploda, folosind o varietate de licențe.

## Arhitectura sistemului de operare: Android

În continuare aș dori să prezint arhitectura *sistemului de operare Android* care cuprinde cinci secțiuni pe patru nivele, acestea fiind: Kernelul Linux, biblioteci, motorul Android și cadrul pentru aplicații.<sup>10</sup>

*Kernelul Linux* utilizează o serie de patch-uri pentru versiune oficială, și conține toate driverele pentru componentele hardware, precum camera, tastatura, WiFi și dispozitive audio. De asemenea, la Kernelul Linux au fost adăugate unele funcționalități specifice Android, acestea fiind: wakelocks, low-memory killer, binder, logger.

*Wakelocks* este o soluție pentru problema de power management folosită de Android, care are ca scop reducerea consumului prin intrarea în starea de "sleep" pentru momentele în care nu este utilizat.

În ceea ce privește *low memory killer*, scopul său este de a opri componentele care nu au fost folosite o perioadă lungă de timp.

Următoarea funcționalitate este reprezentată de *binder*, un mecanism de RPC/IPC, care are ca scop capacitatea de invocare remote a obiectelor asemănătoare cu obiectele COM din Windows.<sup>11</sup>

Și în cele din urmă *logger*, care are ca scop substituirea *sistemului clasic de logging* al kernelului, cu scopul diminuării numărului de task switch-uri și scrieri în fișier, printr-un buffer circular.

O altă componentă este *motorul Android*, alcătuit dintr-o serie de biblioteci de bază, care permit utilizatorilor să dezvolte aplicații mobile, folosind ca limbaj de programare Java; aceste biblioteci permit accesul la funcțiile unui dispozitiv și anume: telefonie, mesaje, gestiunea pachetelor. La baza arhitecturii stau regiștri, fiind echipată cu un compilator JIT, care permite modificarea executabilului obținut pe dispozitivul mobil.

Următoarea componentă este cadrul pentru aplicații: *Android framework*, care furnizează diverse funcționalități ale sistemului de operare pentru ca programatorii să le poată transpune în aplicațiile lor. Această componentă oferă dezvoltatorilor posibilitatea de a realiza aplicații complexe și inovative, întrucât aceștia sunt liberi să utilizeze hardware-ul echipamentelor, de informațiile despre locație, rularea de servicii în background, setarea de alarme, precum și adăugarea de notificări pe bara de stare. Programatorilor li se oferă acces la aceleași API-uri ca și aplicațiile distribuite cu Android, prin urmare arhitectura aplicațiilor este proiectată astfel încât să fie simplificată reutilizarea componentelor: o aplicație poate publica anumite funcționalități, și o altă aplicație să le poată utiliza.

Ultima componentă este reprezentată de *nivelul aplicații*, care oferă atât produsele încorporate în dispozitivele mobile, precum: Camera, Music player, Contacts, și Video player, cât și produsele disponibile pe Play Store.

## Funcționalitățile sistemului de operare Android

Principalele funcționalități pe care sistemul de operare Android le oferă sunt: *stocare*, care folosește SQLite, bază de date relațională ce permite utilizarea eficientă a resurselor; *conectivitatea* prin diverse modalități, precum: 3G, WiFi, Bluetooth, WiMAX, GPRS, EDGE; *WiFi direct*, care permite interconectarea între diverse dispozitive având o lățime de bandă mare; *Android Beam*, prin care utilizatorii partajează conținut instant prin apropierea dispozitivelor respective. O altă funcționalitate este *navigarea pe Internet* bazată pe motorul open source pentru navigare WebKit împreună cu motorul JavaScript de la Chrome V8 suportând HTML5 și CSS3. *Multimedia* admite mai multe formate precum: H.263, M-peg-4, AMR-WEB, AAC, JPEG; *multi-touch*, care suportă posibilitatea de contact în mai multe puncte concomitent; *multi-tasking*;

GCM(Google Cloud Messaging) permițând dezvoltatorilor expedierea de date de dimensiuni reduse, în lipsa unei soluții de sincronizare proprietară.

## Google Maps API

Google a lansat în anul 2005 Google Maps API, care permite programatorilor integrarea de hărților de la Google în propriile aplicații; acest serviciu fiind gratuit. API-ul permite accesul la serverele Google Maps, descărcarea de date, afișarea unei hărți, și trimiterea unui răspuns la interacțiunea cu harta.

O altă facilitare pe care API-ul Google Maps o furnizează este inserarea de informații suplimentare despre obiectele de pe o porțiune a hărții, astfel încât să permită interacțiunea utilizatorului cu harta. Totodată, acesta permite adăugarea de obiecte grafice pe hartă, precum: ancore pentru pozițiile specifice pe hartă, cunoscute sub denumirea de markeri, segmente de linii, segmente închise, imagini și elemente grafice de tip bitmap atașate la poziții specifice pe hartă.

Hărțile afișate de Google Maps API prezintă următoarele caracteristici: titlul acestora nu includ conținut personalizat, iar referitor la pictograme, nu toate permit acțiunea de click. În plus față de funcționalitatea de cartografie, API sprijină o gamă completă de interacțiuni cu harta, în concordanță cu modelul Android UI, un exemplu ilustrativ este posibilitatea configurării interacțiunilor cu harta, prin definirea de *listeneri*, concept care are ca scop oferirea de răspunsuri la gesturile utilizatorilor. Principala clasă care se utilizează în cazul utilizării hărților este *GoogleMap*. Aceasta modelează harta în cadrul aplicației, iar în cadrul interfeței utilizator, harta va reprezentată printr-un *MapFragment* sau *MapView*. Clasa *GoogleMap* permite realizarea următoarelor acțiuni: conectarea la serviciul Google Maps, descărcarea componentelor hărții, afișarea de diverse controale, precum zoom și răspunderea la acțiunea de zoom.

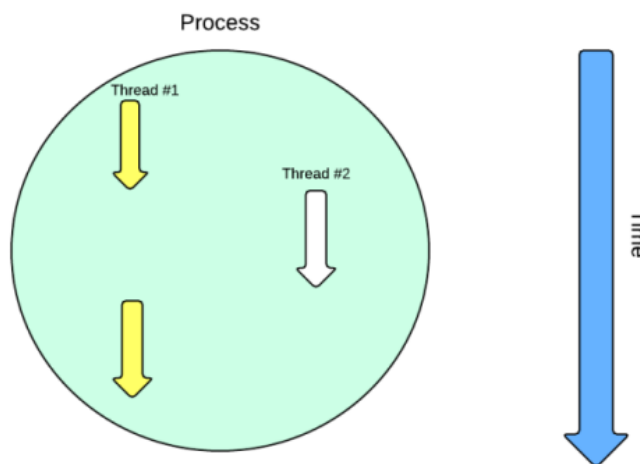
*MapFragment* este o subclasă Android, care permite plasarea unei hărți într-un fragment Android și oferă acces la obiecte de tip *GoogleMap*. Spre deosebire de o extensie de vizualizare (View), un fragment reprezintă un comportament sau o porțiune din interfața utilizator dintr-o activitate, astfel o activitate poate include mai multe fragmente. De asemenea, un fragment poate

fi utilizat în mai multe activități, dar și utilizarea mai multor fragmente pentru realizarea unui interfețe.

În ceea ce privește *MapView* este o subclasă a clasei *Android View*, care permite plasarea hărții într-o extensie de vizualizare. Un *View* reprezintă o regiune a ecranului, element esențial în dezvoltarea de aplicații Android și widget-uri. Mai mult decât atât, prin intermediul unui *MapFragment*, *MapView*-ul acționează asemănător unui container pentru hartă.

## Fire de execuție

Un fir de execuție, numit și *thread*, este o secvență de instrucțiuni care poate fi executată concurent cu o altă secvență, într-un mediu *multithreading*, în timp ce partajează aceeași zonă de memorie. Acest fir de execuție rulează în paralel pentru a crește eficiența programelor. În sistemele cu procesoare multiple sau cu nuclee multiple, firele de execuție rulează în același timp pe procesoare sau nuclee diferite. Pentru procesoarele cu un singur nucleu, sistemul împarte timpul de execuție între *thread*-uri, aspect prezentat în figura de mai jos.



La nivel de implementare se poate utiliza standardul *Pthreads (POSIX Threads)*, ce definește un API pentru crearea și manipularea firelor de execuție. Pentru implementarea unui



server TCP care deservește clienți multipli se utilizează acest standard deoarece aplicațiile ce folosesc fire de execuție sunt în general mai rapide decât dacă utilizează procese.

## SQLite

SQLite este o bibliotecă care implementează o bază de date SQL tranzacțională și care nu are nevoie de o configurație și nici de un server. Codul pentru această librărie este public și poate fi folosit pentru orice scop.

Spre deosebire de alte baze de date SQL, SQLite nu are un server separat pentru procesare. Acesta citește și scrie direct în memoria sistemului. Oferă o bază de date complexă cu posibilitatea de a crea tabele multiple, indici, *trigger*-e și *view*-uri iar tranzacțiile sunt *ACID* (*Consistency, Isolation, Durability*) chiar dacă sunt întrerupte de erori. De asemenea, formatul fișierelor scrise de aceasta bibliotecă este *cross-platform* oferind posibilitatea de a fi utilizat atât pe sisteme de 32 de biți cât și pe cele de 64 sau pe arhitecturi de tip *big-endian* sau *little-endian*. Datorită acestor facilități, SQLite este opțiunea ideală în implementarea unei aplicații performante.

---

<sup>1</sup> <http://www.eseuargumentativromana.com/2016/07/rolul-lecturii-in-dezvoltarea-personala.html>